Report on Massively scalable Sinkhorn distances via the Nyström method

Sebastiano Scardera

1 Abstract

The Sinkhorn distance, a regularized version of the Wasserstein distance, has gained significant popularity in machine learning and statistical inference. However, traditional algorithms for computing this distance suffer from quadratic time and memory complexity with respect to the data size, making them infeasible for large datasets. In the presented paper, we demonstrate that this issue can be effectively addressed by combining two straightforward techniques: the Nyström method and Sinkhorn algorithm. This combination provides an accurate approximation of the Sinkhorn distance while substantially reducing time and memory costs compared to other methods. The analysis, which includes novel insights into the Nyström method and the stability properties of Sinkhorn scaling, supports this result. We also validate our approach through experiments, comparing the accuracy-complexity trade-off with SINKHORN.

2 Introduction

Optimal transport, has recently gained traction in machine learning for applications such as image recognition, domain adaptation, and generative modeling. This growing interest is largely driven by computational advancements, particularly Cuturi's [1] introduction of an entropic regularization, which modifies the optimal transport problem to produce the Sinkhorn distance. This modification significantly reduces computation time while retaining practical utility. Beyond efficiency, the Sinkhorn distance has been shown to exhibit better statistical properties than the unregularized distance.

Let $\mathbf{p}$ and $\mathbf{q}$ be probability distributions supported on at most n points in $\mathbb{R}^d$. Denote $\mathcal{M}(\mathbf{p}, \mathbf{q})$ as the set of couplings between $\mathbf{p}$ and $\mathbf{q}$, and let $H(P)$ represents the Shannon entropy of P. The Sinkhorn distance is defined as:

$$W_\eta(\mathbf{p}, \mathbf{q}) := \min_{P \in \mathcal{M}(p,q)} \sum_{i,j} P_{ij} C_{ij} - \eta^{-1} H(P),$$

where $\eta > 0$ and C_{ij} represents the distance between the point x_i and x_j. In this work it will be the squared Euclidean distance.

Cuturi [1] demonstrated that the Sinkhorn distance can be computed efficiently using Sinkhorn's algorithm, an iterative method that operates on the $n \times n$ cost matrix. While effective for problems of size $n \approx 10^4$, its $O(n^2)$ complexity in both runtime and memory becomes a limitation for larger datasets.

This work contributes to the field of research focused on calculating Sinkhorn distance in a scalable manner.

2.1 Prior work

Computing the Sinkhorn distance efficiently is a well studied problem.

A notable improvement to SINKHORN's performance involved exploring greedy and stochastic variants. For instance, GREENKHORN, proposed by Altschuler et al. [3], is a greedy coordinate descent version of SINKHORN.

Another influential method, introduced by Solomon et al. [2], leverages the fact that when distributions are supported on a grid, SINKHORN scaling can be performed extremely efficiently. This is achieved by decomposing the cost matrix into lower-dimensional slices, leading to decomposable kernels. In our specific case, where $\mathbf{p}$ and $\mathbf{q}$ are distributions on the same set, this approach reduces the complexity to $O(n^{1+1/d})$.

The algorithm presented in this work constructs a low-rank approximation of a Gaussian kernel matrix before applying SINKHORN. The advantage lies in the fact that SINKHORN depends on the kernel only through matrix-vector products. On this line, it is worth highlighting the contributions of Tenetov et al. [6] which computes the approximation via semidiscrete cost approximation and the one of Altschuler et al. [5] that use a Taylor expansion of the Gaussian kernel.

2.2 Paper contribution

This work improve the results of the last two cited paper using an adaptive version of the Nyström approximation. In particular, the results of this paper hold for a generic value of η, while the analyses of Altschuler et al. [5] and Tenetov et al. [6] only yield an approximation to $W_\eta(\mathbf{p}, \mathbf{q})$ when $\eta \to +\infty$.

Furthermore, our algorithm is capable to adapt to the intrinsic dimensionality of the paper that yields a significant improvement in practice.

This work provide independently interesting results on Gaussian kernel approximation via Nyström and stability results about Sinkhorn projections.

Notation Throughout, $\mathbf{p}$ and $\mathbf{q}$ are two probability distributions supported on a set $X := \{x_1, \dots, x_n\}$ of points in $\mathbb{R}^d$, with $x_i \in \mathbb{R}^d$ for all

$i \in [n] := \{1, \dots, n\}$.

We define the cost matrix $C \in \mathbb{R}^{n \times n}$ by $C_{ij} = \frac{1}{2}\|x_i - x_j\|^2$. We identify p and q with vectors in the simplex:

$$\Delta_n := \left\{ v \in \mathbb{R}^n_{\geq 0} : \sum_{i=1}^{n} v_i = 1 \right\},$$

whose entries represent the weights each distribution assigns to the points in X. We denote by $\mathcal{M}(p, q)$ the set of couplings between p and q, identified with the set of matrices $P \in \mathbb{R}^{n \times n}_{\geq 0}$ satisfying $P\mathbf{1} = p$ and $P^\top \mathbf{1} = q$, where $\mathbf{1}$ is the all-ones vector in $\mathbb{R}^n$. The notation $f = O(g)$ means that $f \leq Cg$ for some universal constant C. The notation $\tilde{O}(\cdot)$ omits polylogarithmic factors depending on R, ϵ, n, and δ.

3 Presentation of the method

Our goal is to approximate the Sinkhorn distance with parameter $\eta > 0$:

$$W_\eta(\mathbf{p}, \mathbf{q}) := \min_{P \in \mathcal{M}(\mathbf{p}, \mathbf{q})} \sum_{i,j} P_{ij} \|x_i - x_j\|_2^2 - \eta^{-1} H(P),$$

where $H(P) := \sum_{i,j} P_{ij} \log \frac{1}{P_{ij}}$ denotes the Shannon entropy and we adopt the standard convention that $0 \log \frac{1}{0} = 0$. For shorthand, in the sequel we write

$$V_C(P) := \langle C, P \rangle - \eta^{-1} H(P),$$

for a fixed matrix C. So in our case, we have $W_\eta(\mathbf{p}, \mathbf{q}) = \min_{P \in \mathcal{M}(\mathbf{p}, \mathbf{q})} V_C(P)$. We will indicate with $P^\eta = \operatorname{argmin}_{P \in \mathcal{M}(\mathbf{p}, \mathbf{q})} V_C(P)$. The method that we are going to present is based on Sinkhorn algorithm. We take for granted the results seen in the course on the properties of this algorithm applied to Schrödinger (static) problem.

The authors proposed the Nys-Sink algorithm, which provides an estimate of the Sinkhorn distance by running the Sinkhorn algorithm on a low-rank approximation of the Gibbs kernel. The approximation is provided by a version of the Nyström algorithm using approximate leverage score sampling.

3.1 Classical Nyström algorithm

Given points $X = \{x_1, ..., x_n\}$ with $\|x_i\|_2 \leq R$ for all $i \in [n]$, let $K \in \mathbb{R}^{n \times n}$ denote the matrix with entries $K_{ij} := k_\eta(x_i, x_j)$, where $k_\eta(x, x') := e^{-\eta \|x - x'\|^2}$. For $r \in \mathbb{N}$, we consider an approximation of the matrix K that is of the form

$$\tilde{K} = V A^{-1} V^T,$$

where $V \in \mathbb{R}^{n\times r}$ and $A \in \mathbb{R}^{r\times r}$. In particular, selected a subset $X_r = \{\tilde{x}_1, ..., \tilde{x}_n\} \subset X$ we will consider V and A given by

$$V_{ij} = k_\eta(x_i, \tilde{x}_j), \quad A_{ij} = k_\eta(\tilde{x}_i, \tilde{x}_j).$$

NYS-SINK will depends on $\tilde{K}$ only through matrix-vector products that can be computed efficiently as

$$\tilde{K}v = V((L^{-1})^T(L^{-1}(V^T v))),$$

where $A = LL^T$ is the Cholesky decomposition of A (that costs $O\left(r^3\right)$) and each product of the form $L^{-1}v$ can be efficiently computed solving the triangular linear system $Lx = v$, with a computational cost $O(nr)$.

3.2 Proposed algorithm

Algorithm 1 NYS-SINK

Input:$X = \{x_1, ..., x_n\} \subseteq \mathbb{R}^d$, $\mathbf{p}, \mathbf{q} \in \Delta_n$, $\epsilon, \eta > 0$
Output: $\hat{P} \in \mathcal{M}(\mathbf{p}, \mathbf{q}), \hat{W} \in \mathbb{R}, r \in \mathbb{N}$

1: $\epsilon' \leftarrow \min\left(1, \frac{\epsilon\eta}{50(4R^2 n + \log\frac{n}{\eta\epsilon})}\right)$
2: $(\tilde{K}, r) \leftarrow$ ADAPTIVENYSTRÖM$(X, \eta, \frac{\epsilon'}{2}e^{-4nR^2})$ ▷ Compute low-rank approximation
3: $(D_1, D_2, \hat{W}) \leftarrow$ SINKHORN$(\tilde{K}, \mathbf{p}, \mathbf{q}, \epsilon')$ ▷ Approximate Sinkhorn projection and cost
4: $\hat{P} \leftarrow$ ROUND$(D_1\tilde{K}D_2, \mathbf{p}, \mathbf{q})$ ▷ Round to feasible set
5: **return** $\hat{P}, \hat{W}$

As we can see from 1, NYS-SINK involves several subroutines:

- ADAPTIVENYSTRÖM: searches for the lowest possible (fixed precision threshold) rank Nyström approximation. To do this it uses the **doubling trick**(line 3, 2) which makes it adaptive.

  Data sampling in Nyström is done using the **approximate ridge leverage scores** computed by Algorithm 2 of [4]. With respect to uniform sampling, the ridge leverage score gives more weight to data that are "less dependent" on the others, resulting in more meaningful sampling.

- SINKHORN: is the classical version computed on the approximated kernel.

- ROUND: starting from Sinkhorn's result, it returns a feasible solution. It can be proved that the resulting matrix lies in the ℓ_1-ball with centre the original matrix and ray the ℓ_1-error on the marginal distributions of the original matrix.

Algorithm 2 AdaptiveNyström

Input: $X = \{x_1, ..., x_n\} \subseteq \mathbb{R}^d$, $\tau, \eta > 0$
Output: $\tilde{K} \in \mathbb{R}^{n \times n}, r \in \mathbb{N}$

```
err ← +∞, r ← 1
while err > τ do
    r ← 2r                                  ▷ Doubling trick
    K̃ ← Nyström(X, η, r)                    ▷ with leverage-score sampling
    err ← 1 − min_{i∈[n]} K̃_ii
end while
return (K̃, rank(K̃))
```

3.3 Theoretical guarantees

3.3.1 Main result

Nys-Sink generates a low-rank Nyström approximation by dynamically determining the minimal rank needed, tailored to the dataset's characteristics. The key quantity for understanding the error of this algorithm is the so called *effective dimension* (also sometimes called the degrees of freedom) of the kernel K.

Definition 3.1. Let $\lambda_j(K)$ denote the jth largest eignevalue of K (with multiplicity). Then the *effective dimension* of K at level $\tau > 0$ is

$$d_{\text{eff}}(\tau) := \sum_{j=1}^{n} \frac{\lambda_j(K)}{\lambda_j(K) + \tau n}.$$

The effective dimension $d_{\text{eff}}(\tau)$ indicates how large the rank of an approximation $\tilde{K}$ to K must be to give the guarantee $\|K - K\|_{op} \leq \tau n.$. For our application it is sufficient to find an approximate kernel $\tilde{K}$ which satisfies $\|K - K\|_{op} \leq \frac{\epsilon'}{2} e^{-4\eta R^2}$, where $\epsilon' = \tilde{O}(\epsilon R^{-2})$. So it is worth defining the following quantity

Definition 3.2. Given $X = \{x_1, ..., x_n\} \subseteq \mathbb{R}^d$ with $\|x_i\| \leq R$ for all $i \in [n]$, $\eta > 0$, and $\epsilon' \in (0, 1)$, the *approximation rank* is

$$r^*(X, \eta, \epsilon') := d_{\text{eff}} \left(\frac{\epsilon'}{2n} e^{-4\eta R^2} \right),$$

where $d_{\text{eff}}(\cdot)$ is the effective rank of the kernel matrix $K := e^{-\eta C}$.

The main result is the following.

Theorem 3.3. *Let* $\epsilon, \delta \in (0, 1)$. Nys-Sink *runs in* $\tilde{O}\left(nr\left(r + \frac{\eta R^4}{\epsilon}\right)\right)$ *time, uses O(n(r+ d)) space, and returns a feasible matrix* $\hat{P} \in \mathcal{M}(\mathbf{p}, \mathbf{q})$ *in fac-*

tored form and scalars $\hat{W} \in \mathbb{R}$ *and* $r \in \mathbb{N}$*, where*

$$\begin{aligned} \left|V_C(\hat{P}) - W_\eta(\mathbf{p}, \mathbf{q})\right| &\leq \epsilon \\ KL\left(\hat{P} \| P^\eta\right) &\leq \eta\epsilon \\ \left|\hat{W} - W_\eta(\mathbf{p}, \mathbf{q})\right| &\leq \epsilon \end{aligned}$$

and, with probability 1 - δ,

$$r \leq c \cdot r^*\left(X, \eta, \epsilon'\right) \log \frac{n}{\delta},$$

for a universal constant c *and where* $\epsilon' = \Omega\left(\epsilon R^{-2}\right)$*.*

We note that the running time is controlled by r^*. One can show that r^* adapts to the intrinsic dimension of the data. In particular, approximation properties of the Gaussian kernel over manifolds gives a bound on the effective dimension of the kernel. Let $\Omega \subset \mathbb{R}^d$ be a smooth compact manifold without boundary, and $k < d$, with atlas $(\Psi_j, U_j)_{j\in[T]}$, with $T \in \mathbb{N}$. We will work under the following smoothness assumption.

Assumption 1. *There exists* $Q > 0$ *such that*

$$\sup_{u \in B^k_{r_j}} \left\| D^\alpha \Psi_j^{-1}(u) \right\| \leq Q^{|\alpha|}, \quad \alpha \in \mathbb{N}^k, j \in [T],$$

where $|\alpha| = \sum_{j=1}^k \alpha_j$ *and* $D^\alpha = \frac{\partial^{|\alpha|}}{\partial u_1^{\alpha_1} \ldots \partial u_k^{\alpha_k}}$*.*

Theorem 3.4. *Let* $\Omega \subset B^d_R \subset \mathbb{R}^d$ *be a smooth compact manifold without boundary satisfying Assumption 1. Let* $X \subset \Omega$ *be a set of cardinality* $n \in \mathbb{N}$*.*

Let $\tau \in (0, 1]$*,* K *the Gaussian kernel matrix associated to* X *and* $d_{\mathit{eff}}(\tau)$ *the effective dimension computed on* K*. Then, there exists* c_1, c_2 *not depending on* $X, n,$ *or* τ*, for which*

$$d_{\mathit{eff}}(\tau) \leq \left(c_1 \log \frac{1}{\tau}\right)^{5k/2} + c_2.$$

As a consequence, using the definition we also get that there exists $c_{\Omega,\eta} > 0$ *not depending on* X *or* n *such that*

$$r^*(X, \eta, \epsilon') \leq c_{\Omega,\eta} \left(\log \frac{n}{\epsilon'}\right)^{5k/2}.$$

Corollary 3.5. *If* X *lies on a suitable* k*-dimensional manifold , then with high probability* Nys-Sink *requires* $\tilde{O}\left(n \cdot \frac{c_{\Omega,\eta}}{\epsilon}\left(\log \frac{n}{\epsilon}\right)^{5k}\right)$ *time.*

To achieve this results we need for new results on Nyström approximation of Gaussian kernel and new stability results for Synkhorn projections.

3.3.2 Adaptive Nyström guarantees

One can show that, with high probability, Gaussian kernels can be approximated by Nyström and in particular by AdaptiveNyström arbitrarly well with a rank controlled by the effective dimension. It holds the following

Lemma 3.6. *Let $(\tilde{K}, r)$ denote the (random) output of* AdaptiveNyström(X, η, τ). *Then:*

1. $\left\|K - \tilde{K}\right\|_\infty \leq \tau$;
2. *The algorithm used $O(nr)$ space and terminated in $O(nr^2)$ time;*
3. *There exists a universal constant c such that simultaneously for every $\delta > 0$,*

$$\mathbb{P}\left(r \leq c \cdot d_{\mathit{eff}}\left(\frac{\tau}{n}\right)\log\left(\frac{n}{\delta}\right)\right) \geq 1 - \delta.$$

3.3.3 Approximate Sinkhorn scaling

Algorithm 3 Sinkhorn

Input: $\tilde{K}$ (in factored form), $\mathbf{p}, \mathbf{q} \in \Delta_n$, $\delta > 0$
Output: Positive diagonal matrices $D_1, D_2 \in \mathbb{R}^{n\times n}$, cost $\hat{W}$

$\tau \leftarrow \frac{\delta}{8}$, $D_1, D_2 \leftarrow I_{n\times n}$, $k \leftarrow 0$
$\mathbf{p}' \leftarrow (1-\tau)\mathbf{p} + \frac{\tau}{n}\mathbf{1}$, $\quad \mathbf{q}' \leftarrow (1-\tau)\mathbf{q} + \frac{\tau}{n}\mathbf{1}$ ▷ Round $\mathbf{p}$ and $\mathbf{q}$
while $\|D_1\tilde{K}D_2\mathbf{1} - \mathbf{p}'\|_1 + \|(D_1\tilde{K}D_2)^\top\mathbf{1} - \mathbf{q}'\|_1 > \frac{\delta}{2}$ **do**
  $k \leftarrow k + 1$
  if k is odd **then**
    $(D_1)_{ii} \leftarrow \mathbf{p}'_i/(\tilde{K}D_2\mathbf{1})_i$ for $i = 1, \ldots, n$ ▷ Normalize rows
  else
    $(D_2)_{jj} \leftarrow \mathbf{q}'_j/((D_1\tilde{K})^\top\mathbf{1})_j$ for $j = 1, \ldots, n$
  end if
end while
$\hat{W} \leftarrow \sum_{i=1}^n \log(D_1)_{ii}((D_1\tilde{K}D_2)\mathbf{1})_i + \sum_{j=1}^n \log(D_2)_{jj}(((D_1\tilde{K}D_2)^\top\mathbf{1})_j)$
return $D_1, D_2, \hat{W}$

Recall that V_C is defined as $V_C(P) := \langle M, P\rangle - \eta^{-1}H(P)$. For any cost matrix C, it can be shown that $V_C(\cdot)$ exhibits stability in the following cases: (i) when performing Sinkhorn projection on an approximate kernel matrix $\tilde{K}$, and (ii) when using an approximate Sinkhorn projection.

Additionally, a runtime bound can be established as a function of the matrix-vector product cost associated with $\tilde{K}$.

Theorem 3.7. *If $K = e^{-\eta C}$ and if $\tilde{K} \in \mathbb{R}^{n\times n}_{>0}$ satisfies $\left\|\log K - \log\tilde{K}\right\|_\infty \leq \epsilon'$, then* Sinkhorn *in* Nys-Sink *outputs D_1, D_2, and $\hat{W}$ such that $\tilde{P} := D_1\tilde{K}D_2$ satisfies*

1. $\left\|\tilde{P}\mathbf{1} - \mathbf{p}\right\|_1 + \left\|\tilde{P}^T\mathbf{1} - \mathbf{q}\right\|_1 \leq \epsilon'$ *and* $\hat{W} = V_{\tilde{C}}(\tilde{P})$;

2. $\left|V_C(P^\eta) - V_C(\tilde{P})\right| \leq \frac{\epsilon}{2}$ *and* $\left|\hat{W} - V_C(\tilde{P})\right| \leq \frac{\epsilon}{2}$

Moreover, if matrix-vector products can be computed with $\tilde{K}$ and $\tilde{K}^T$ in time $\mathrm{T}_{\mathrm{MULT}}$, then this takes time $O\left((n + \mathrm{T}_{\mathrm{MULT}})\eta \left\|C\right\|_\infty \epsilon'^{-1}\right)$.

4 Experimental results

To run our experiments, we used Google Colab with an Intel Xeon CPU @ 2.20GHz, 2 cores, and 12GB RAM (standard runtime). We want to plot the time-accuracy trade-off for Nys-Sink for several ranks, compared to the standard Sinkhorn algorithm.

We can notice that the version of Nys-Sink described in the previous section in the form of Algorithm 1 doesn't allow to choose the rank of the approximation. For this reason we implemented a slightly different version as in the Algorithm 4.

Algorithm 4 Iterative Nys-Sink

Input:$X = \{x_1, ..., x_n\} \subseteq \mathbb{R}^d$, $\mathbf{p}, \mathbf{q} \in \Delta_n$, $\eta > 0$, $r, \mathrm{N}_{\mathrm{iter}} \in \mathbb{N}$
Output: $\hat{P} \in \mathcal{M}(\mathbf{p}, \mathbf{q}), \hat{W} \in \mathbb{R}$

1: $(V, L) \leftarrow$ Nyström(X, η, r) ▷ Compute low-rank approximation
2: $(D_1, D_2, \hat{W}) \leftarrow$ Sinkhorn$(V, L, \mathbf{p}, \mathbf{q}, \mathrm{N}_{\mathrm{iter}})$ ▷ Approximate Sinkhorn projection and cost
3: $\hat{P} \leftarrow$ Round$(V, L, D_1, D_2, \mathbf{p}, \mathbf{q})$ ▷ Round to feasible set
4: **return** $\hat{P}, \hat{W}$

The two main difference with Nys-Sink is that we don't use anymore the doubling trick and Sinkhorn is implemented with a fixed number of iterations. In line 1, we implemented Nyström with approximate ridge leverage-score sampling. To implement the sampling we use Algorithm 2 of [4] whose code is provided by the authors. In Algorithm 4 we made it explicit that along the algorithm $\tilde{K}$ is in factored form. Since both Sinkhorn and Round depend on $\tilde{K}$ only through matrix-vector product, we implemented them in the efficient way described in the Subsection 3.1.

Accuracy-complexity evaluation To evaluate the accuracy of the methods we solved the Entropic regularized problem with CVXPY, and we use this solution to evaluate the excess risk of Nys-Sink and Sinkhorn. To estimate the complexity we computed the number of FLOPs needed for each iteration (up to constant) for the two methods ($O(nr)$ for Nys-Sink and $O(n^2)$ for Sinkhorn, note that it coincides with the cost of the matrix-vector product with the kernel), so that for each iteration we know approximately

how many operations have been performed so far. The curves are shifted to the right by the cost of the initialisation ($O(nr^2)$ for Nys-Sink and $O(n^2)$ for Sinkhorn).

Experimental setup This experiment is performed on a uniform distribution of data with value in $[0, 0.8)$ of size n $= 1000$, which corresponds to cost matrices of dimension approximately 1000×1000. We construct the distribution $\mathbf{p}$ starting from a uniform distribution on the points, doubling the weights for the first half and renormalising. For $\mathbf{q}$ it is the same but we have doubled the weights for the second half. As cost matrix we use $C_{ij} = \|x_i - x_j\|^2$ and the corresponding kernel $K_{ij} = e^{-\eta C_{ij}}$. We have fixed $\mathrm{N}_{\mathrm{iter}} = 40$ for the Sinkhorn algorithm and we do not do the rounding in line 2 of the algorithm 3.

The time-accuracy trade-off was analyzed for Nys-Sink using ranks $r = 25, 50, 75, 100, 125, 150$. These values were chosen to be approximately one order of magnitude smaller than n. We conducted our experiments by testing various values of η ($\eta = 5, 15, 25, 30$) across different data dimensions $d = 2, 3, 5$.

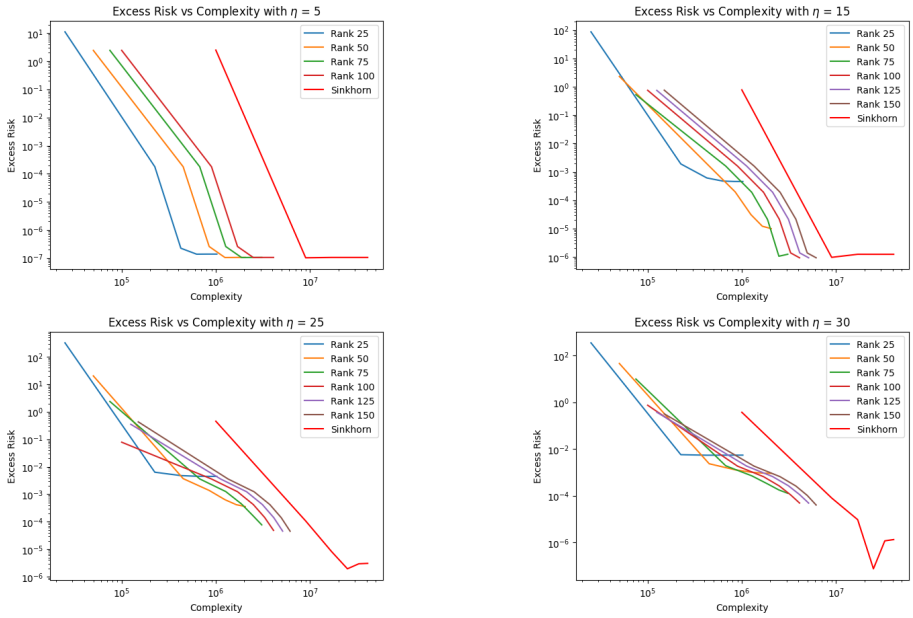

Figure 1: Time-accuracy tradeoff for Nys-Sink and Sinkhorn in dimension 2, for a range of regularization parameters and approximation ranks r

Comments on Results The low-rank approximation provides good results in dimensions 2 and 3, especially when a higher regularization parameter (η smaller) is used. However, in higher dimensions, such as 5D,

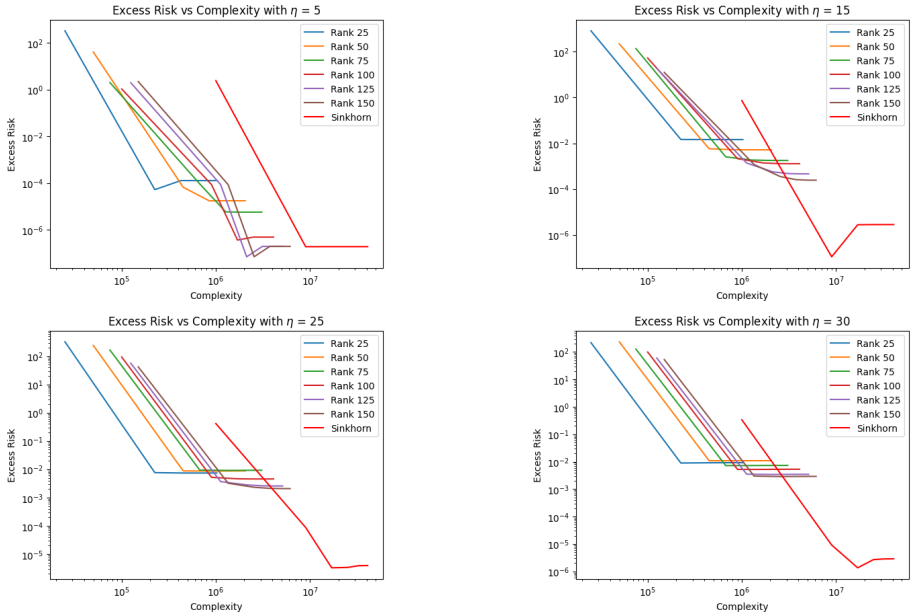

Figure 2: Time-accuracy tradeoff for NYS-SINK and SINKHORN in dimension 3, for a range of regularization parameters and approximation ranks r

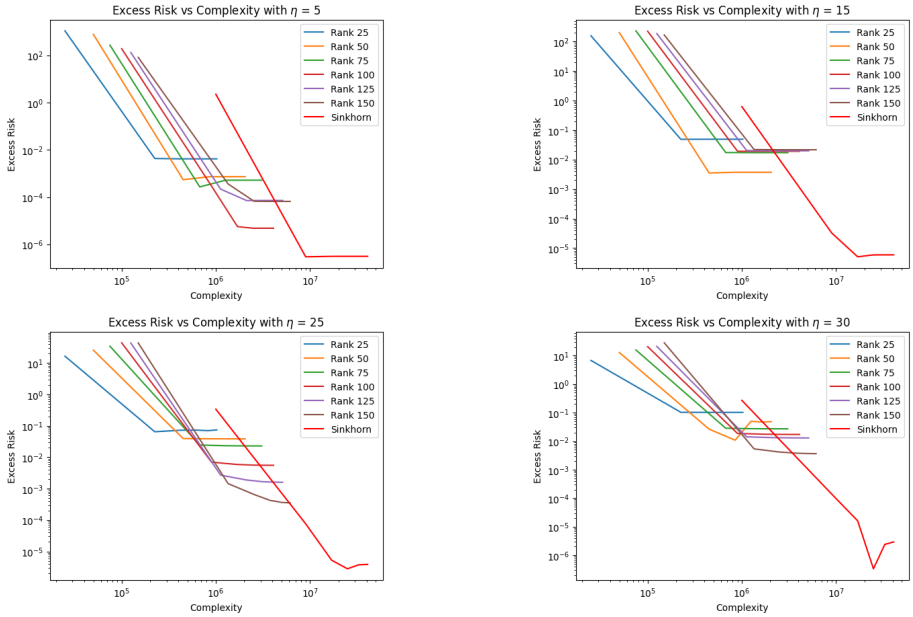

Figure 3: Time-accuracy tradeoff for NYS-SINK and SINKHORN in dimension 5, for a range of regularization parameters and approximation ranks r

the limitations become more apparent. This may be due to the fact that $n = 1000$ is insufficiently expressive for such high-dimensional spaces.

In terms of accuracy, SINKHORN, in the unlimited budget setting, consistently achieves the best results. Although it reaches convergence in just a few iterations, these iterations are computationally expensive, making SINKHORN much more costly than the other methods. In dimensions 3 and 5, under the same setting, we observe that NYS-SINK with higher rank can achieve better results. This phenomenon does not appear in our experiments for $d = 2$.

Positive-definite issues During the experimental phase we encountered some problems with the positive definite of the approximate kernel. In particular, this problem occurred for η was large (so the regularisation parameter was small), when r was too large (for a fixed d) or vice versa when d was too small (for a fixed r). In particular, the first plot of the first figure of the experiments for $r = 125, 150$ is missing for this reason. We would also like to experiment the method in dimension 1, but it would only work with too small a value of r, making it incomparable with the other dimension.

5 Critics

The results on NYS-SINK are valid only when C represents the squared Euclidean distance, which produces a Gaussian kernel. This limitation arises from the kernel approximation step via Nyström, where the Gaussianity is explicitly utilized, either through a Taylor expansion or in the context of RKHS.

Several limitations emerge when we put in practice the methods. In particular, numerical issues related to the positive definiteness of the approximation were observed in certain scenarios, as highlighted in the final paragraph of Section 4, causing the method to fail.

The authors propose a method with a complexity of $o(n^2)$, provided the matrix is maintained in its factored form. However, if the full matrix needs to be computed, the complexity becomes $\approx n^2$, which can be infeasible for large values of n.

As the authors admitted, instead of using Algorithm 2 to adaptively choose the rank, they conducted experiments with a fixed and relatively small choice of r. We would have appreciated additional experiments demonstrating the original NYS-SINK algorithm, particularly showcasing its adaptive capabilities.

The last critics is about a minor typo. In the pseudocode of SINKHORN presented in the original paper, we believe there is an incorrect inequality in line 3.

6 Conclusion and perspective

The authors introduce a novel method NYS-SINK for approximating the Sinkhorn distance in a fast and reliable manner, combining Nyström method and Sinkhorn Algorithm. Moreover, NYS-SINK is able to adapt to the intrinsic dimension of the data, something new in this argument.

To substantiate their claims, they provide both runtime guarantees and error bounds. Their main contributions include deriving new results on Gaussian kernel approximation using the Nyström method and establishing stability properties for the objective function of the entropic regularized transport problem.

This article opens the door to several potential future works:

- Developing more general results by attempting to approximate a broader class of kernels.
- Investigating sharper constants for the bounds in the Gaussian kernel case.
- Understanding why NYS-SINK performs well even when r is smaller than the theoretical guarantees.

7 Connection with the course

NYS-SINK addresses the regularized optimal transport problem, also referred to as the Schrödinger (static) problem. The method builds on SINKHORN but incorporates a low-rank approximated kernel, differing from the standard version discussed in the course.

The computational complexity of SINKHORN is Tn^2, where T denotes the number of iterations. By employing the approximated kernel, the complexity reduces to Trn, as detailed in Subsection 3.1.

The stability of Sinkhorn projections depends on conditions that recall the distance that makes Sinkhorn operator contractive and so SINKHORN convergent.

References

[1] Marco Cuturi. "Sinkhorn Distances: Lightspeed Computation of Optimal Transport". In: *Advances in Neural Information Processing Systems*. Ed. by C.J. Burges et al. Vol. 26. Curran Associates, Inc., 2013. URL: https://proceedings.neurips.cc/paper_files/paper/2013/file/af21d0c97db2e27e13572cbf59eb343d-Paper.pdf.

[2] Justin Solomon et al. "Convolutional wasserstein distances: Efficient optimal transportation on geometric domains". In: *ACM Transactions on Graphics (ToG)* 34.4 (2015), pp. 1–11.

[3] Jason Altschuler, Jonathan Niles-Weed, and Philippe Rigollet. "Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/491442df5f88c6aa018e86dac21d3606-Paper.pdf.

[4] Cameron Musco and Christopher Musco. "Recursive sampling for the nystrom method". In: *Advances in neural information processing systems* 30 (2017).

[5] Jason Altschuler et al. "Approximating the quadratic transportation metric in near-linear time". In: *arXiv preprint arXiv:1810.10046* (2018).

[6] Evgeny Tenetov, Gershon Wolansky, and Ron Kimmel. "Fast entropic regularized optimal transport using semidiscrete cost approximation". In: *SIAM Journal on Scientific Computing* 40.5 (2018), A3400–A3422.